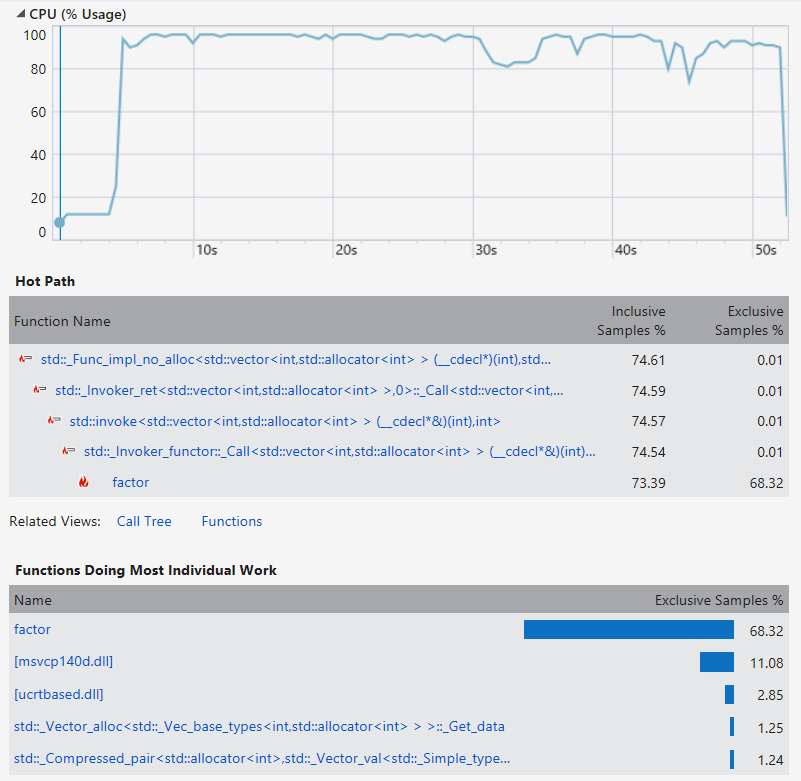
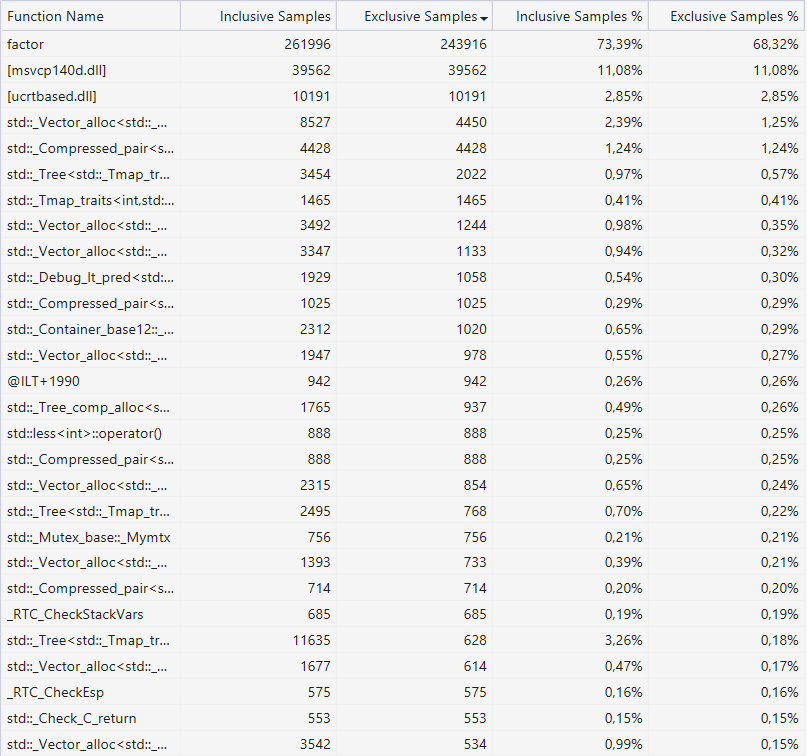
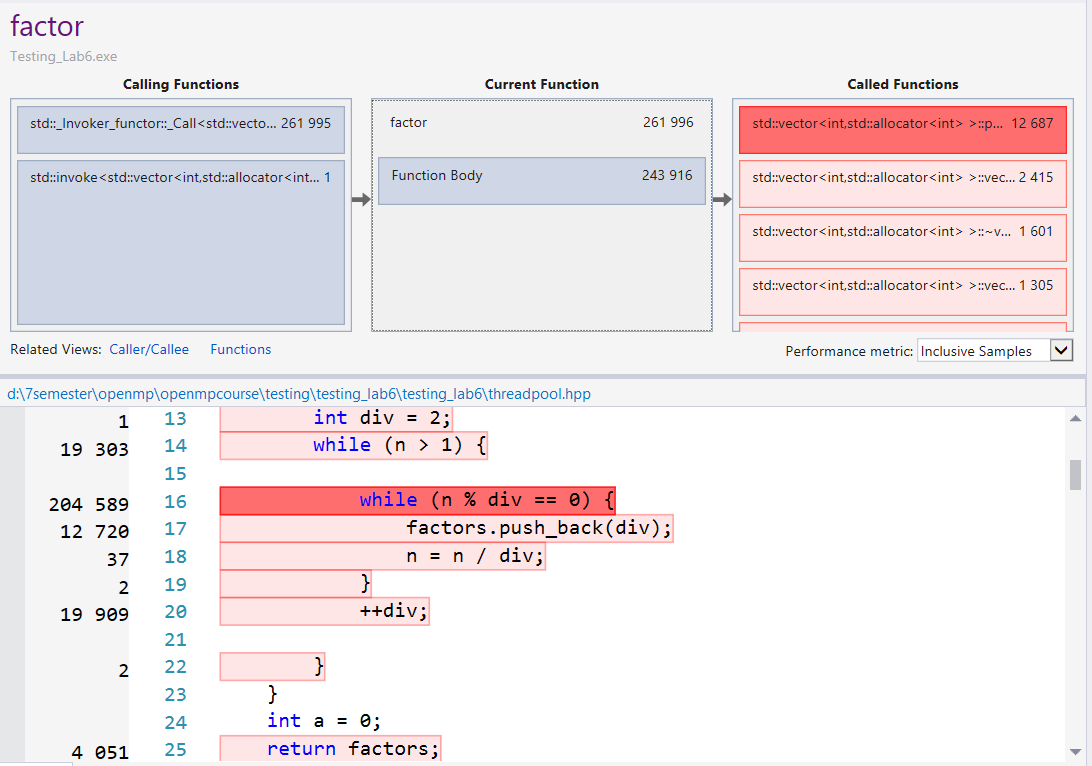
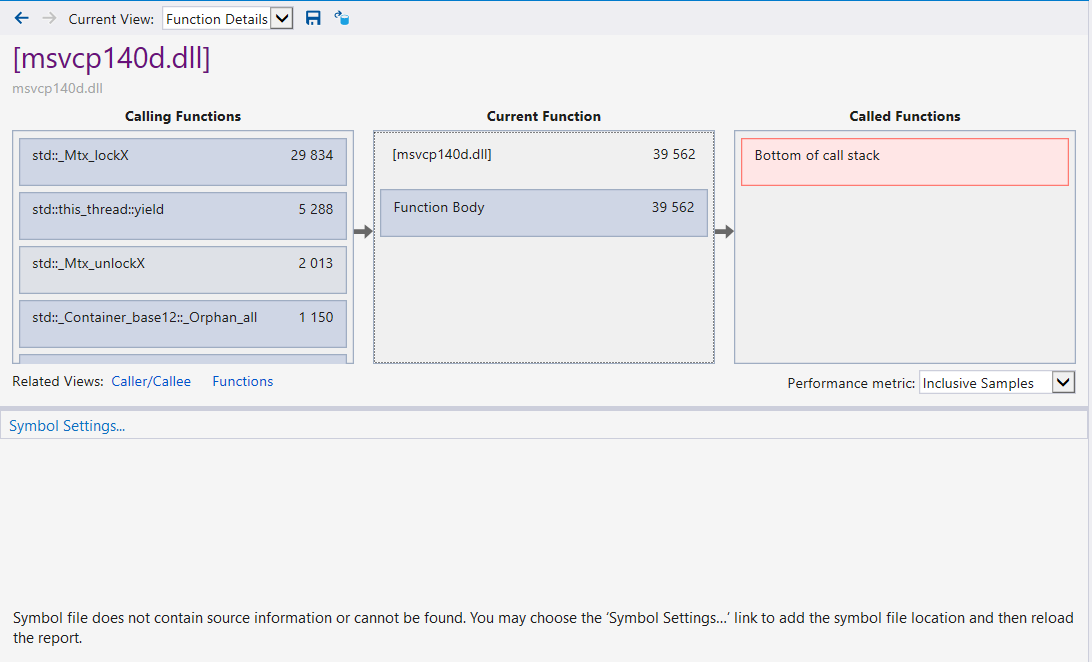
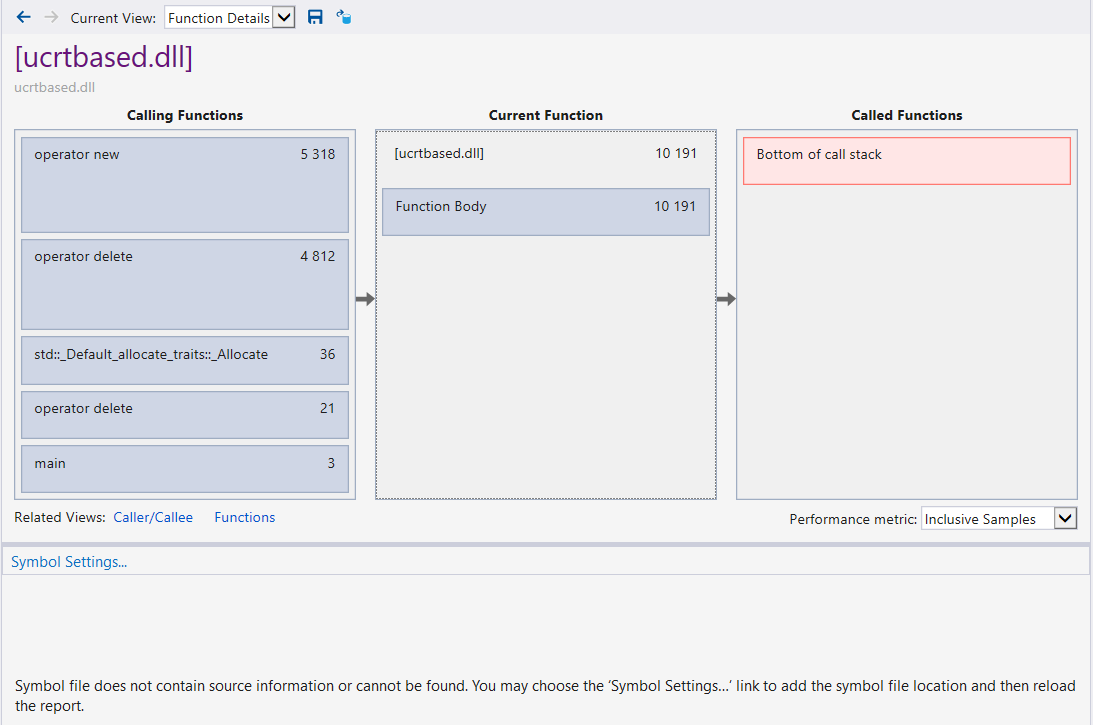
1. Провести профилирование с помощью **дискретного профилировщика**. В отчет занести результаты из представлений Summary, Functions. Для функций, на выполнение которых затрачено больше всего процессорного времени, добавить в отчет информацию из представления Function Details. Для данных в каждом из этих представлений написать комментарии.

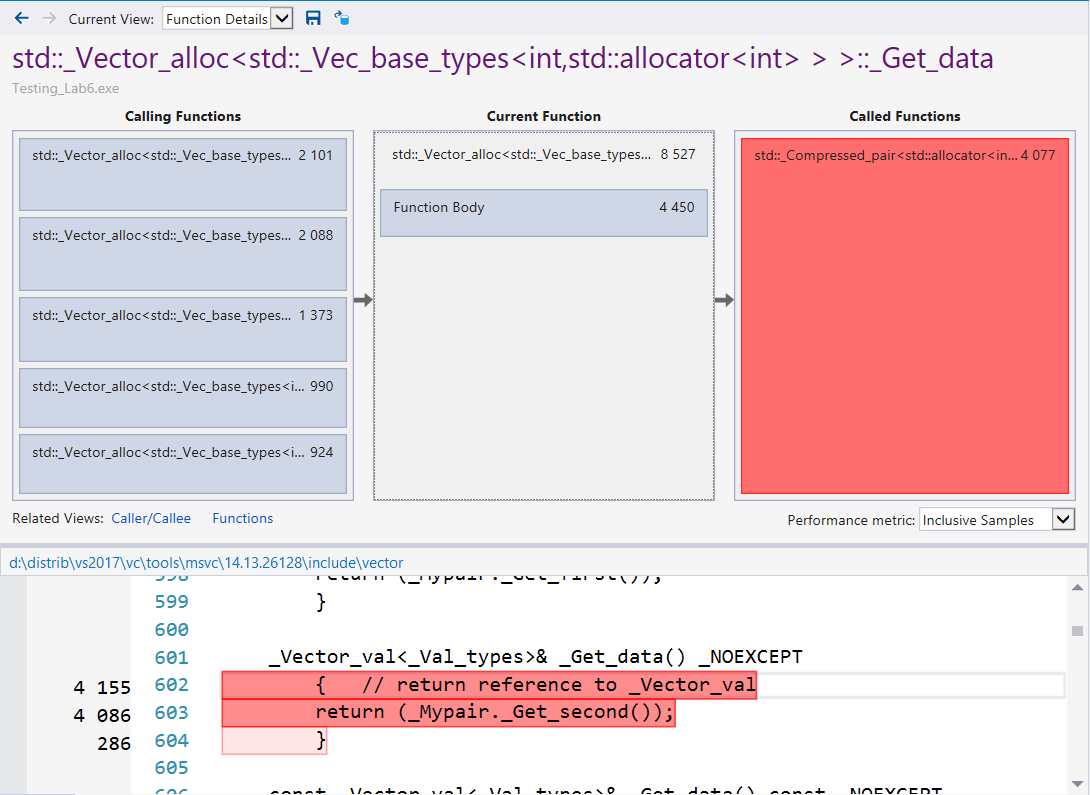




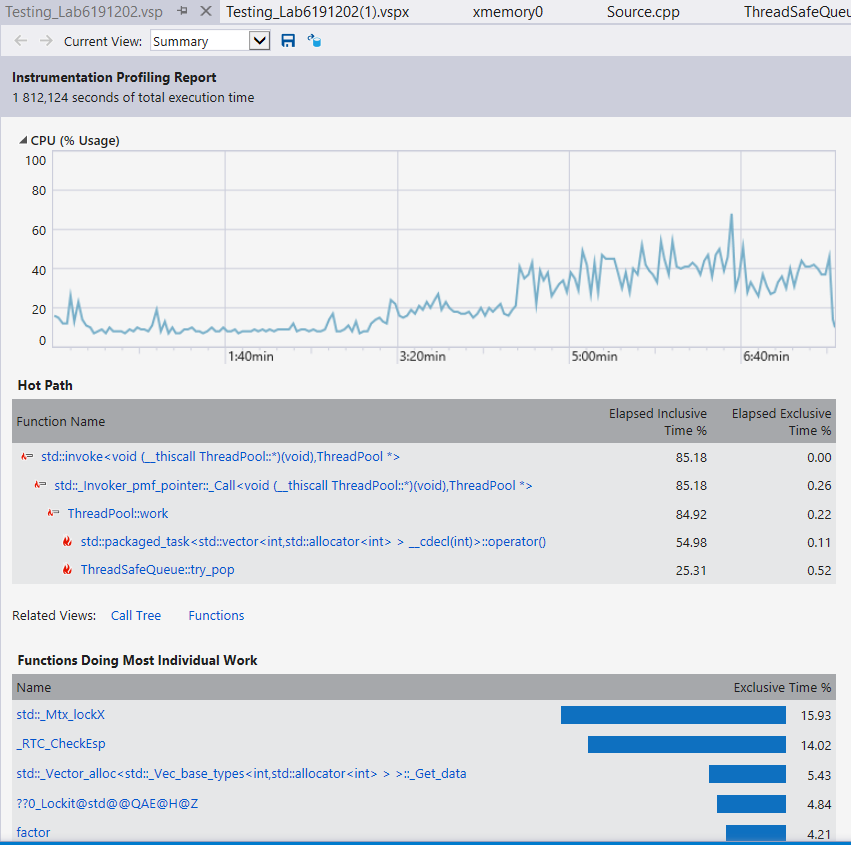


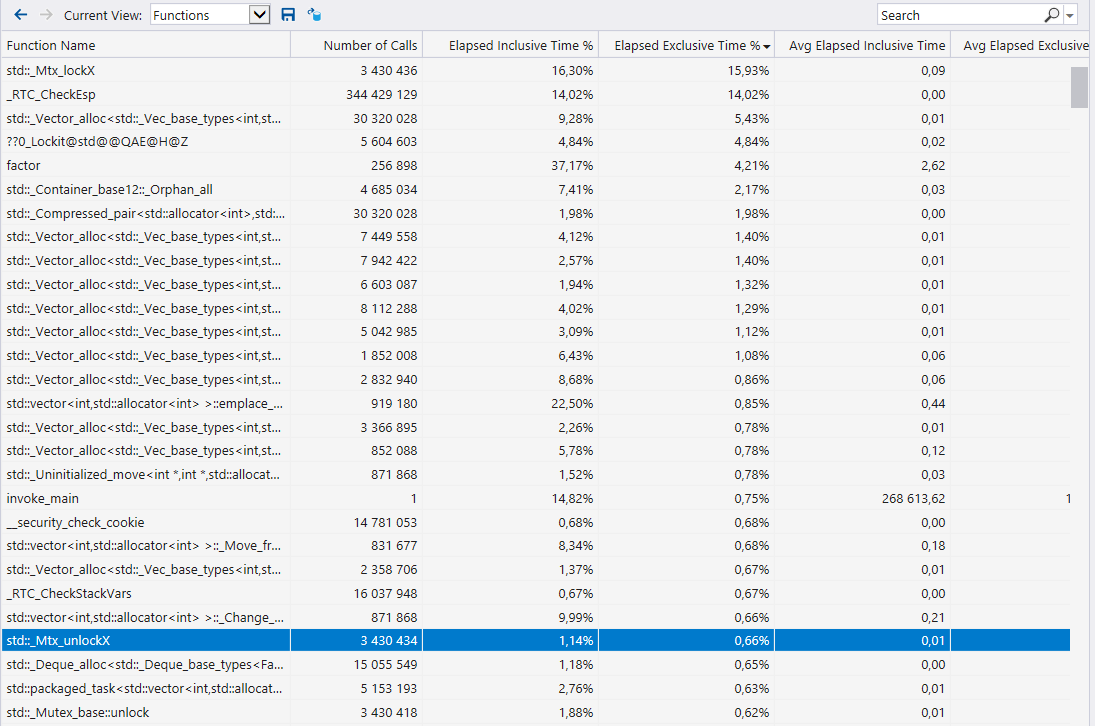


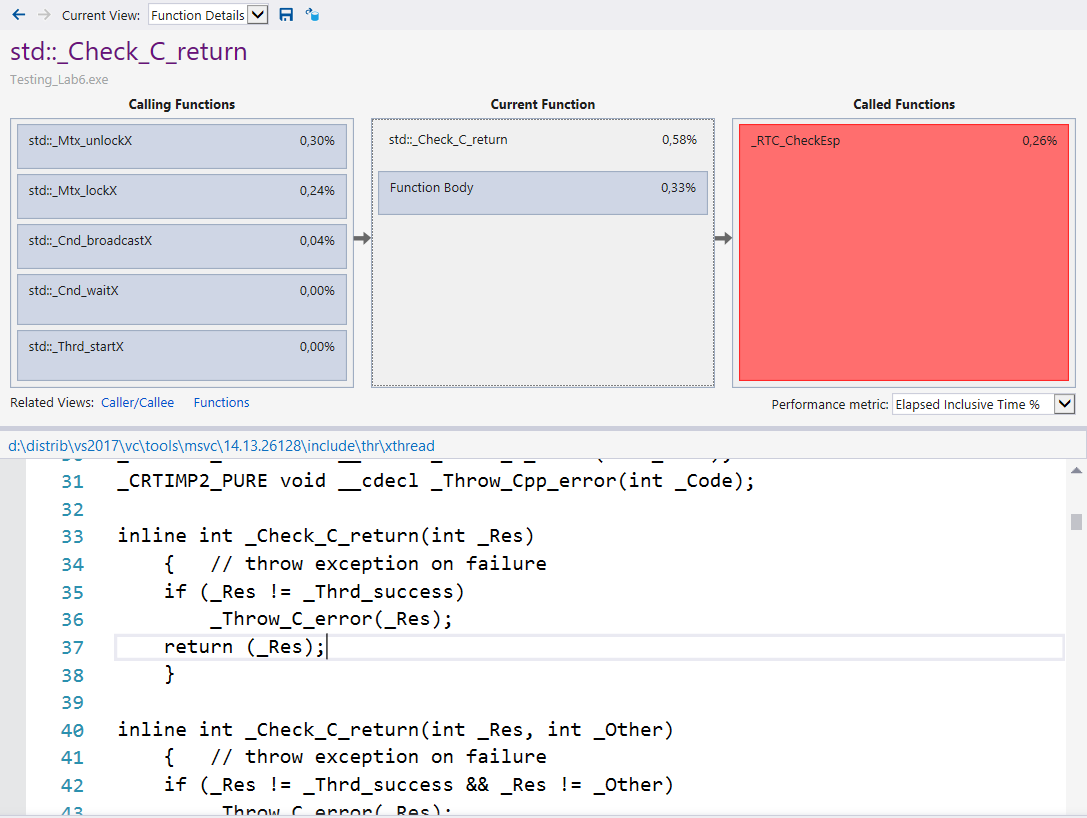


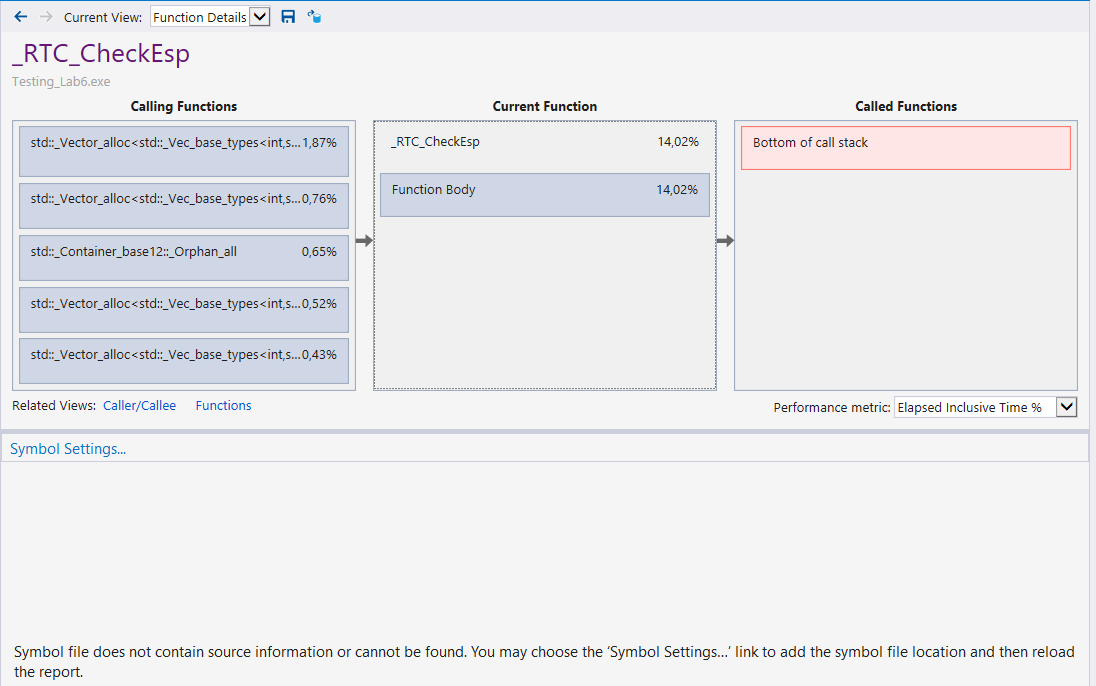


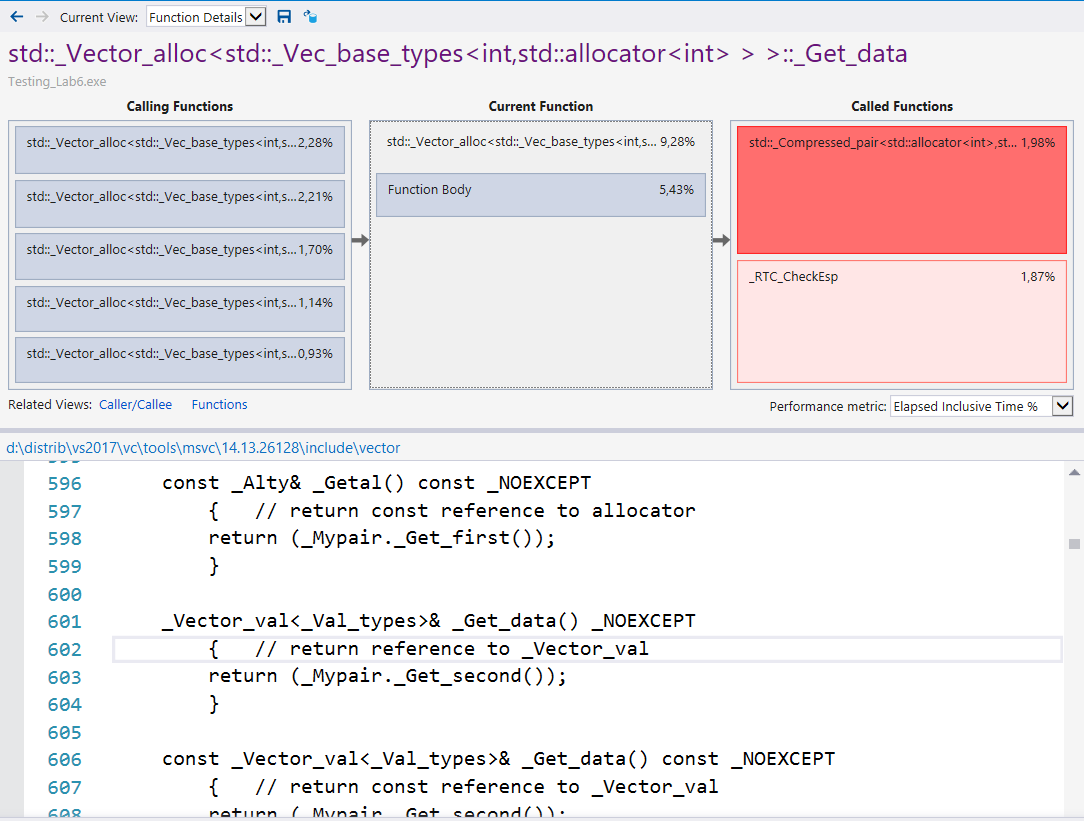
1. Провести профилирование с помощью **инструментированного профилировщика**. В отчет занести результаты из представлений Summary, Functions. Для функций, у которых наибольшее исключительное время, добавить в отчет информацию из представления Function Details. Для данных в каждом из этих представлений написать комментарии.



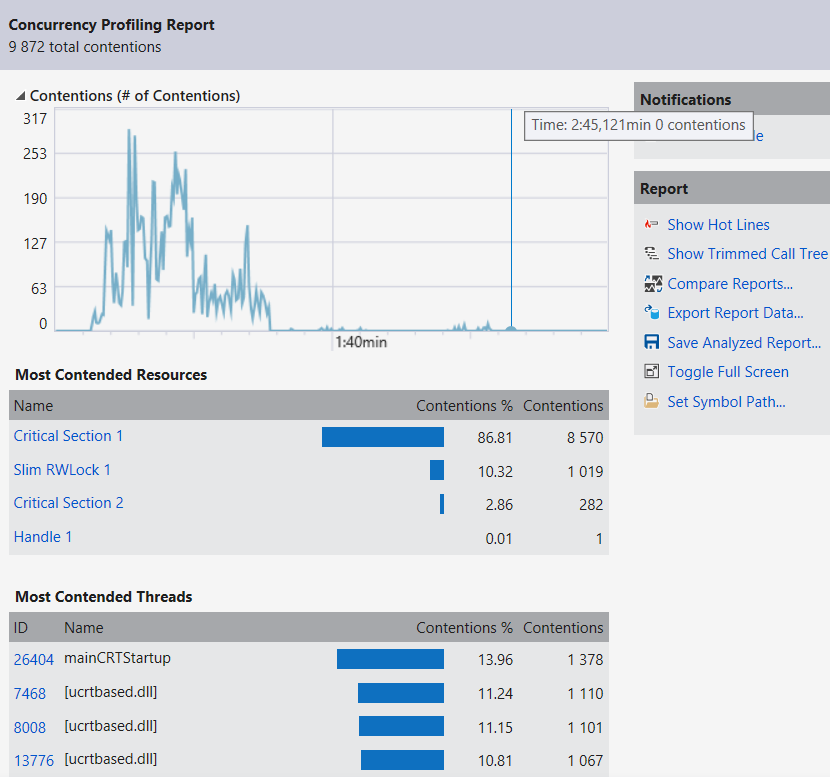


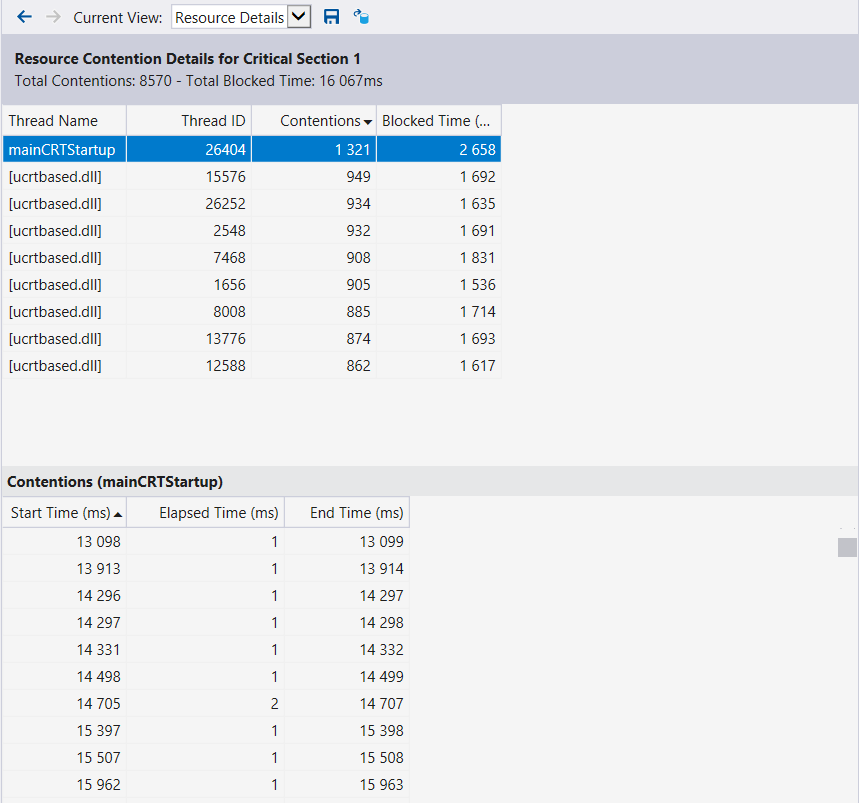


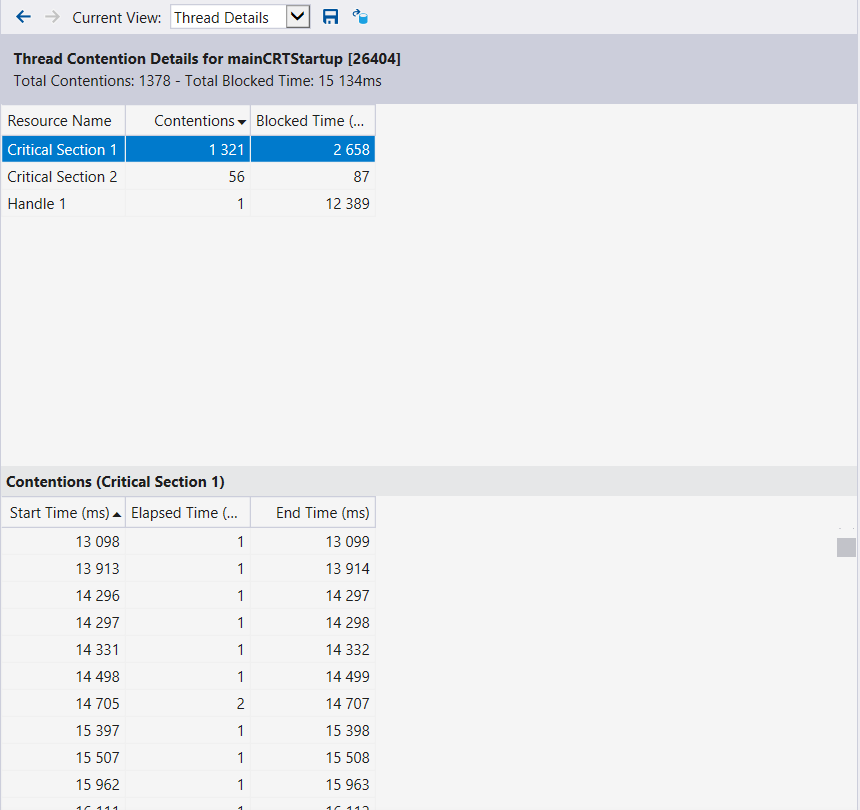




1. Провести профилирование с помощью **профилировщика выделения памяти**. В отчет занести результаты из представлений Summary, Functions. Для функций, которые выделили больше всего памяти (исключительное значение), добавить в отчет информацию из представления Function Details. Также в отчет включить информацию из представления Allocation, а именно какие ветви в стеке вызовов выделили памяти больше всего. Для данных в каждом из этих представлений написать комментарии.
2. Провести профилирование с помощью **профилировщика конкуренции**. Для режима Concurrency в отчет занести результаты из представления Resource Details, Threads Details. Для режима Concurrency Visualizer – сводную информацию. Добавить комментарии.







1. На основании полученной информации сделать вывод о причинах временных затрат, большого выделения памяти и т.п. при выполнении программы. Можно собрать информацию и о других событиях, если это требуется для обоснования вывода.